

Table of Contents

Abstract.....	i
Acknowledgement.....	ii
Chapter 1 Project Definition.....	1
Chapter 2 Description.....	3
Chapter 3 Software and Hardware Requirements.....	5
3.1 Hardware Requirements.....	5
3.2 Software Requirements.....	5
Chapter 4 Major Functionality.....	7
4.1 Property Search.....	7
4.2 View Properties.....	7
4.3 Property Details.....	7
Chapter 5 Activity Chart.....	9
Chapter 6 Screenshots of your project output.....	11
6.1 Home Page.....	11
6.2 Property Search Results.....	13
6.3 Property Details Page.....	14
6.4 Rental and Buy Property Page.....	15
Chapter 7 Limitations of project.....	17
Chapter 8 Outcome.....	19
Chapter 9 Future Enhancement.....	21
Chapter 10 Conclusion.....	24
References.....	26

ABSTRACT

This application aims to enhance emergency preparedness, response, and recovery by providing crucial features for users during natural disasters. The app integrates functionalities like offline accessibility, an AI-powered weather companion for real-time forecasts and alerts, and an SOS system for quick emergency assistance. It also includes a contact list for emergencies and push notifications to keep users informed about impending disasters. The technology stack incorporates Flutter SDK for cross-platform development, and real-time data management, and APIs like Twilio for communication services and Novu for notifications. Google Cloud Platform and Gemini API are utilized for weather data and scalable infrastructure. The app's core focus is on ensuring reliability even in disconnected environments, providing a comprehensive tool for safety, communication, and information during critical times. By leveraging modern technologies, this app aspires to offer an effective solution for disaster response and management.

ACKNOWLEDGEMENT

I, the developer of "Femme Secure," a comprehensive real estate application, with immense pleasure and commitment, would like to present this project assignment. The development of this project has provided me with a wide array of opportunities to think critically, implement solutions effectively, and interact with various aspects of web development and emerging technologies.

I hereby take this opportunity to express my heartfelt gratitude to several individuals who generously extended their valuable time, support, and cooperation in developing this project.

I express my deep sense of gratitude towards our Head of the Computer Engineering Department, Dr. Dweepna Garg, for their unwavering support and guidance throughout the entire duration of this project. I am profoundly thankful to my project guide, Prof. Premal Patel, whose expert advice, insightful feedback, and encouragement were instrumental in the successful completion of this project. Their support prompted me to work diligently and adopt new technologies, which significantly enhanced the quality of my work.

Thank you,

Probin Bhagchandani (22DCE006)

Shivaansh Dave (22DCE017)

Japan Kachhiya (22DCE040)

Vraj Shah (23D2DCE150)

CHAPTER 1: Project Definition

Project Definition

Description

In today's world, natural disasters such as earthquakes, hurricanes, floods, and wildfires are becoming more frequent and severe, often causing widespread damage and loss of life. Timely information, efficient communication, and preparedness are key factors in mitigating the impact of these disasters. To address these needs, this disaster management application provides a comprehensive and user-friendly solution for disaster preparedness, response, and recovery. The app is designed to function reliably in critical situations, even when offline, ensuring users have access to vital information and communication tools during emergencies.

CHAPTER 2: Description

Project Objectives:

- Enhance preparedness by providing users with real-time weather updates, forecasts, and disaster alerts through an AI-powered weather companion.
- Enable swift emergency response through an SOS feature that sends alerts to emergency contacts and services during critical situations.
- Support reliable communication during disasters by offering offline functionality, allowing users to access important information even without an internet connection.
- Provide essential safety information through push notifications, detailing emergency protocols and actions users should take during specific disaster events.
- Facilitate recovery efforts by connecting users with local authorities and community resources, as well as providing a platform for post-disaster updates and assistance.

This project's innovation lies in its integration of modern technology to build an accessible and reliable disaster management tool, aiming to reduce the impact of natural disasters on individuals and communities.

CHAPTER 3: Software and Hardware Requirements

Software and Hardware Requirements.

3.1 Hardware Requirements

Processor: Any modern processor capable of running a web browser.

RAM: 4GB or more for optimal performance.

Hard Drive: No specific requirement, as the application is web-based and does not require local storage.

Display: Any monitor or screen with a resolution of 1024x768 or higher.

3.2 Software Requirements

Operating System: Any modern operating system that supports the latest web browsers (Windows 10, macOS, Linux).

Web Browser: Google Chrome, Mozilla Firefox, Safari, Microsoft Edge, or any other modern web browser with JavaScript enabled.

Development Tools: Not required for end-users, but for developers, tools like Visual Studio Code, Atom, or Sublime Text can be used.

Frameworks and Libraries: The application may use frontend frameworks like ReactJS, AngularJS, or Vue.js, which are loaded from content delivery networks (CDNs) and do not require installation on the end-user's machine.

CHAPTER 4: Major Functionality

Major Functionality

1. AI Weather Companion

Delivers real-time weather updates and disaster forecasts, leveraging AI to predict potential natural calamities such as hurricanes, floods, and wildfires. Users receive personalized weather alerts based on their location, helping them stay informed and take proactive measures to prepare for emergencies.

2. SOS Emergency Alerts

The SOS feature allows users to send distress messages with their current location to predefined emergency contacts and services. Integrated with Twilio API, this ensures rapid transmission of emergency signals, even in low network conditions, offering users a reliable method to request help during critical situations.

3. Offline Access

The app ensures that vital information like evacuation routes, emergency contacts, and disaster preparedness guides are accessible offline. Users can navigate important safety information even when internet or cellular networks are unavailable, making it a crucial tool during network outages or disasters.

4. Push Notifications

Sends real-time disaster alerts, evacuation orders, and safety protocols through push notifications, keeping users updated about ongoing situations. These notifications are personalized based on the user's location, ensuring that they receive the most relevant and timely information to act quickly.

5. Emergency Contact List

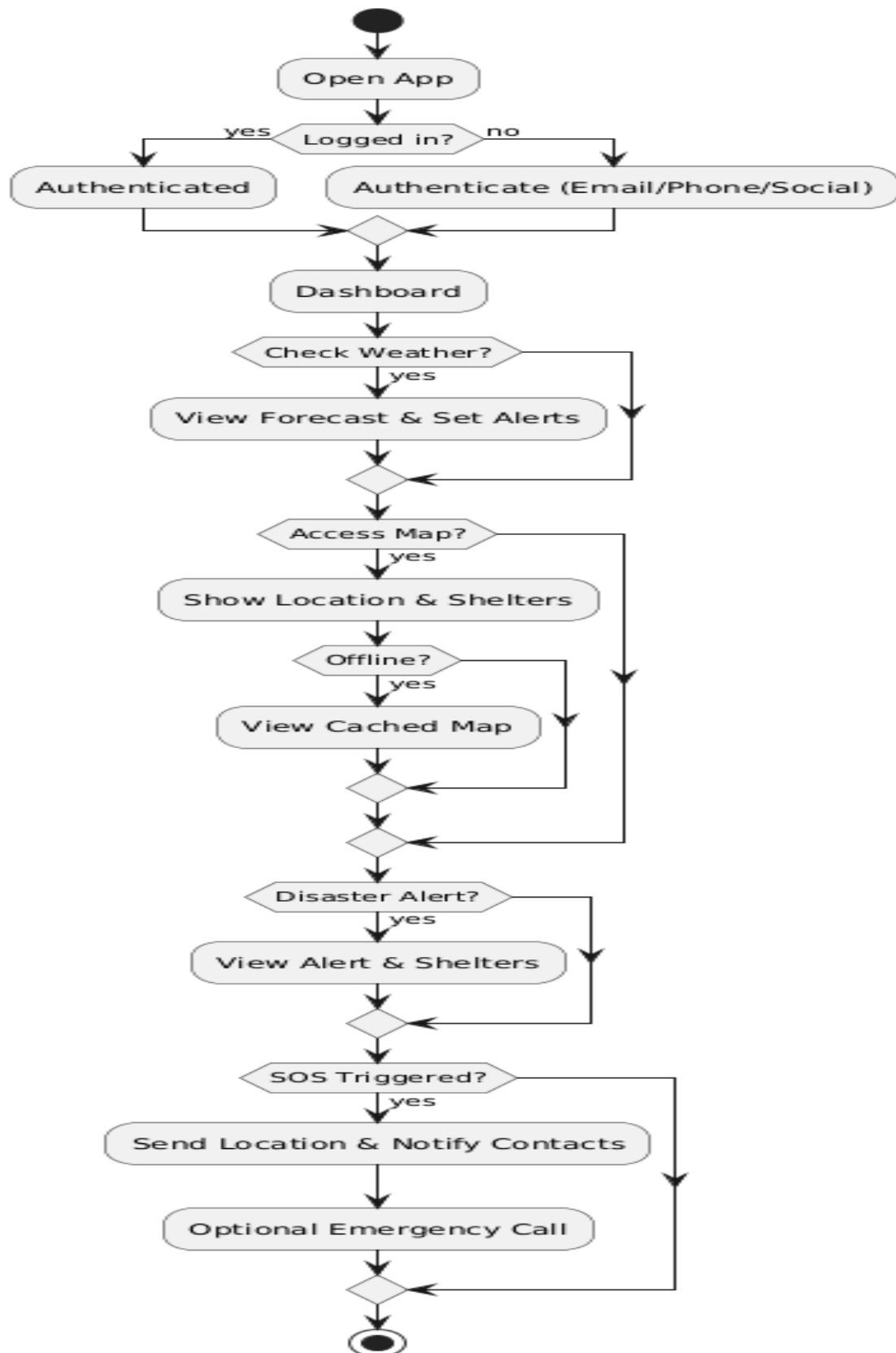
Users can store and organize key emergency contacts, such as family members, local authorities, or emergency services. During a disaster, users can quickly communicate with these contacts, ensuring fast and reliable communication when time is critical.

6. Resource Locator

Provides users with a real-time map of nearby resources, including shelters, medical camps, and relief centers. By integrating with Google Maps, the app helps users locate essential services and navigate to safety during or after a disaster, improving their chances of finding aid.

CHAPTER 5: Flow Chart

Flow Chart



Preparations for the project:

Process 1: Learning about Technology This phase involves understanding the project's core concepts and objectives, studying relevant technologies and reviewing existing real estate platforms for inspiration.

Process 2: Analyzing the Requirements In this chapter, user needs and project requirements are identified. Market research and competitive analysis are conducted to finalize the project's specifications and features.

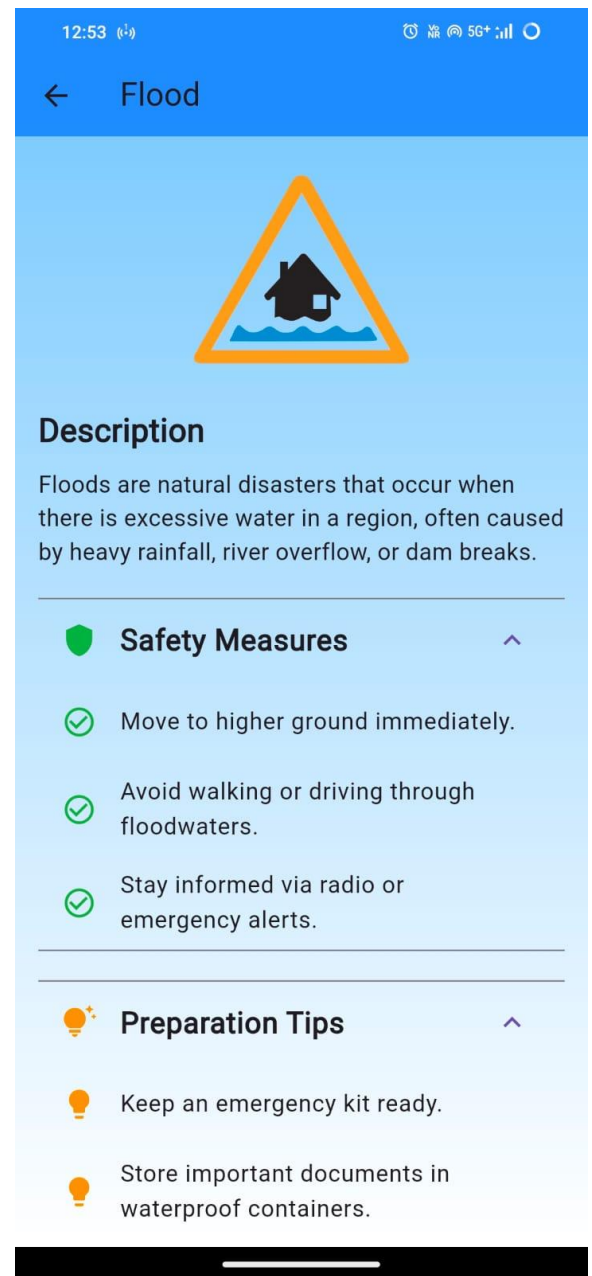
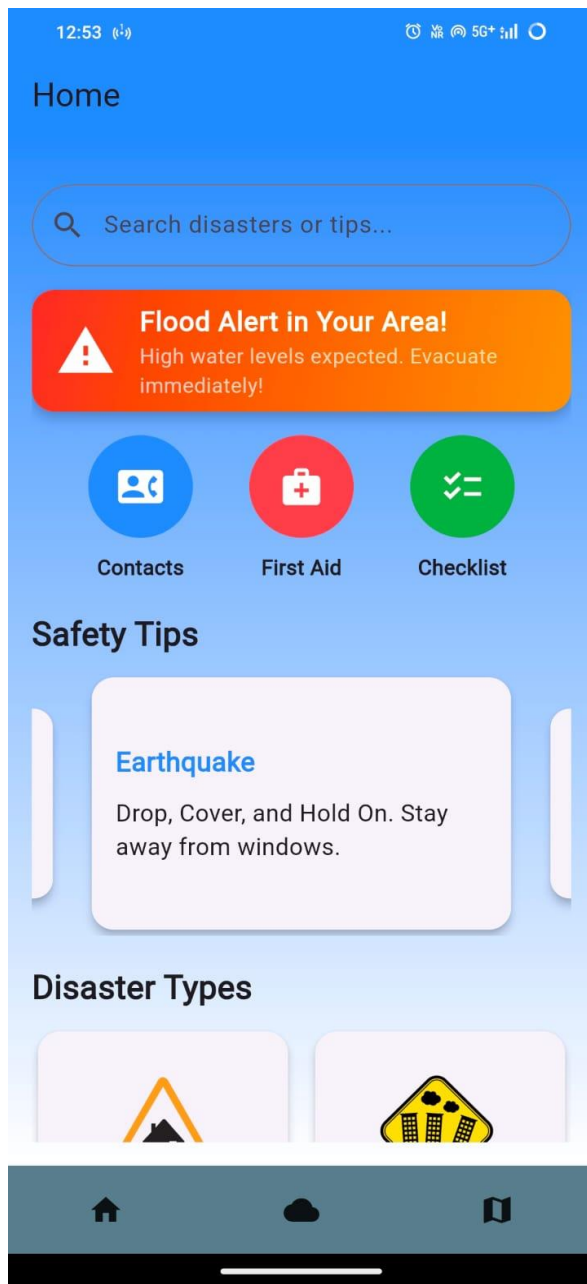
Process 3: Designing the Project This stage focuses on creating prototype designing for the key pages, and refining the designs based on feedback to ensure an intuitive and engaging user experience.

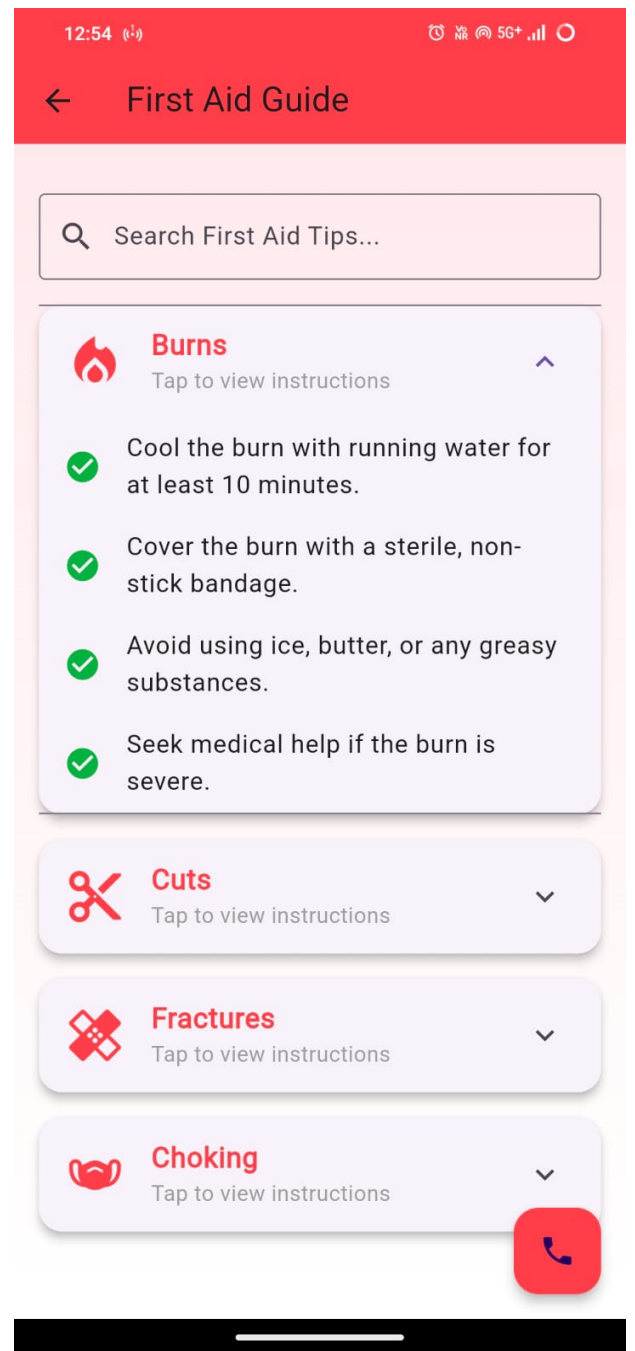
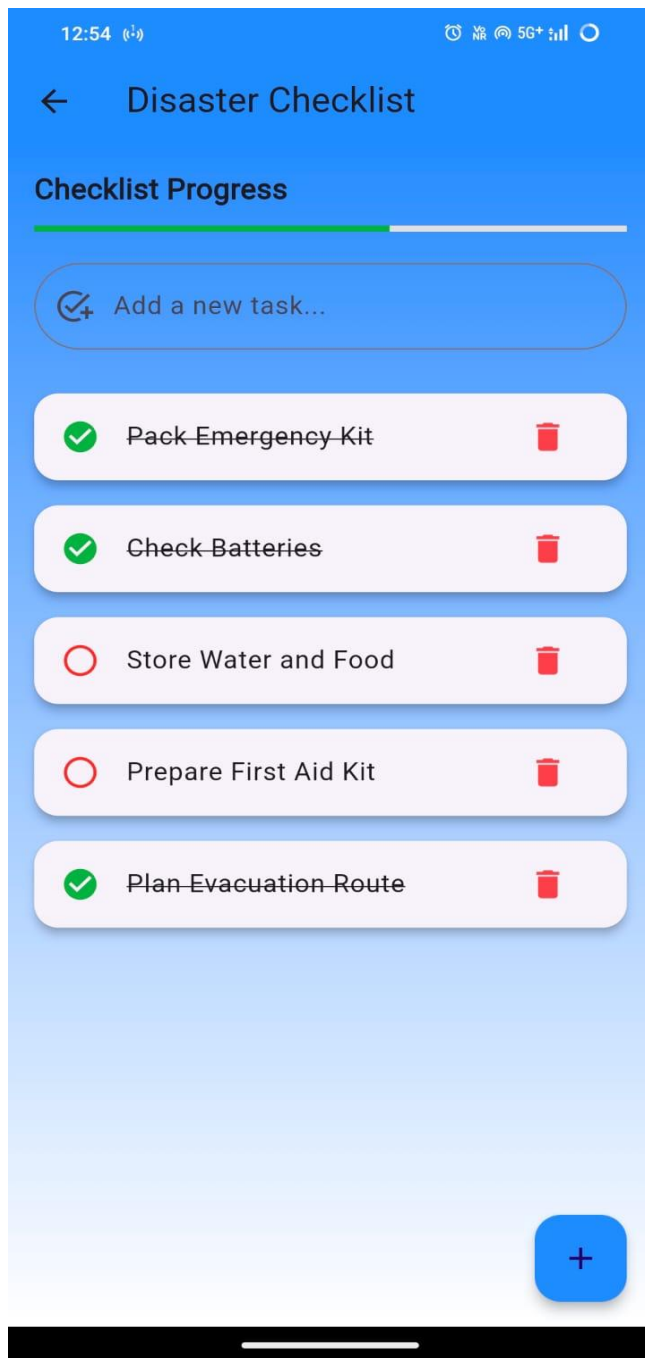
Process 4: Implementation and Coding Here, the development environment is set up, core functionalities (such as Property Search and View Properties) are developed, and Property Details are integrated with a responsive design.

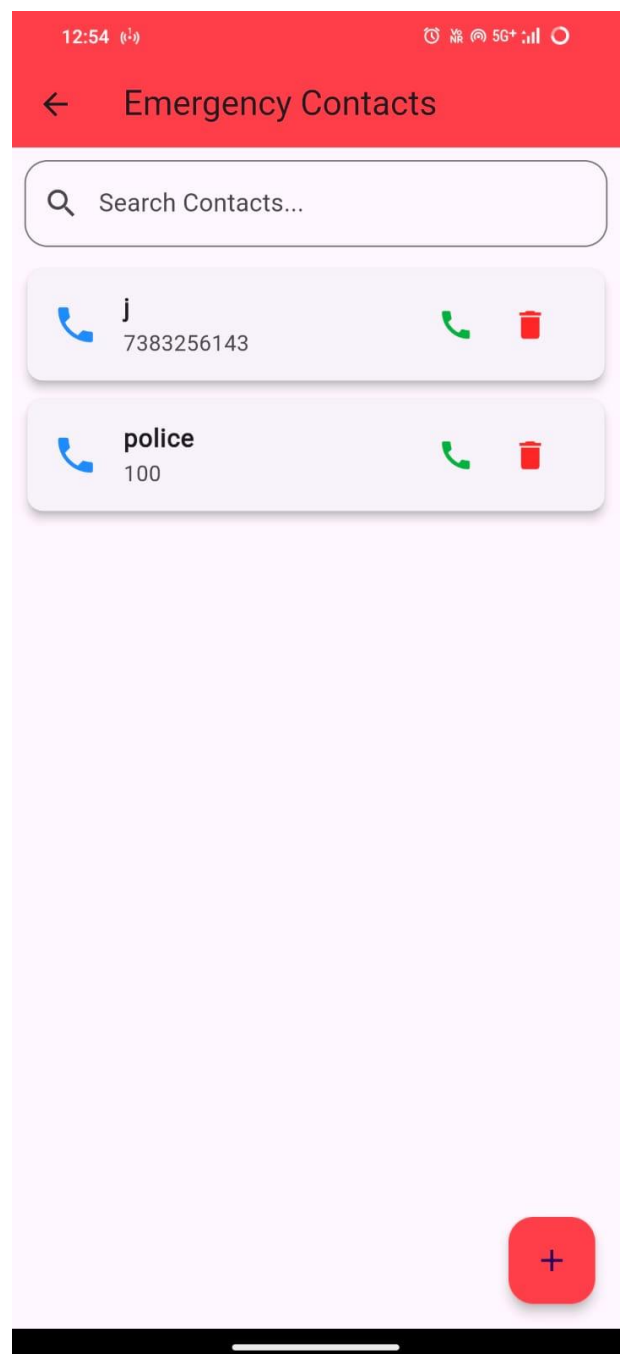
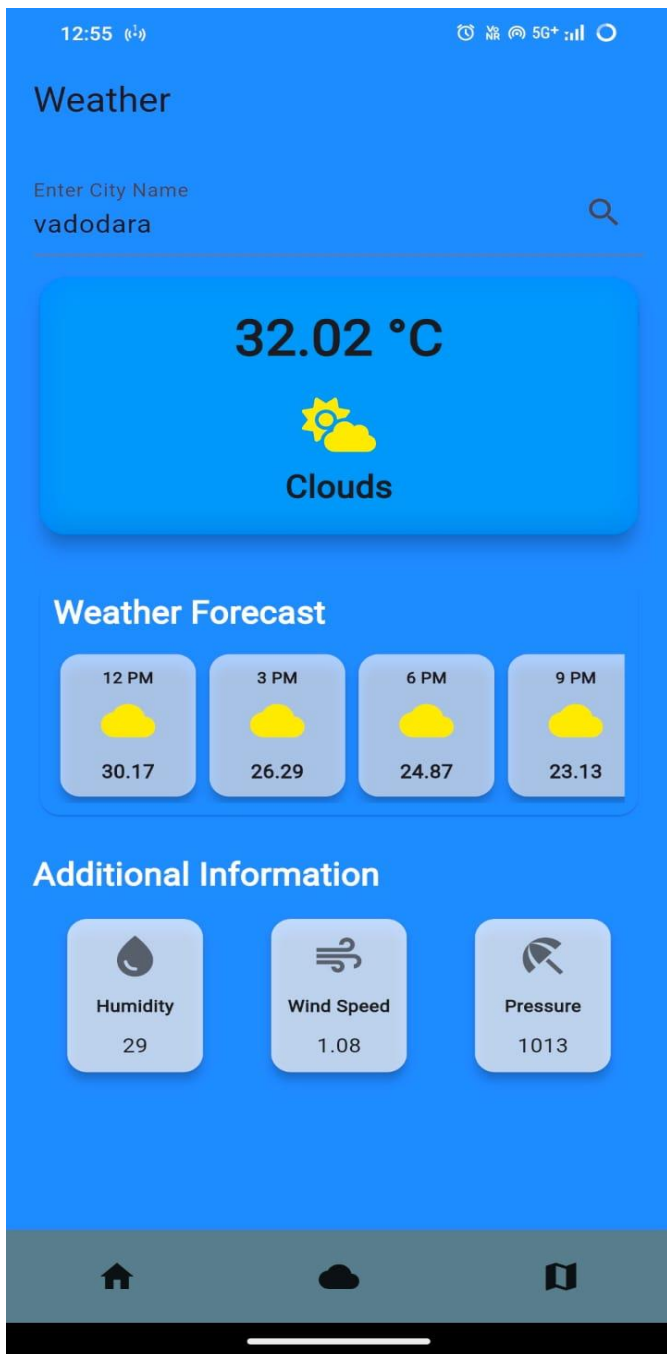
Process 5: Testing and Validating The final phase involves conducting unit and integration testing, followed by user testing to gather feedback and make final adjustments, ensuring the project is user-friendly.

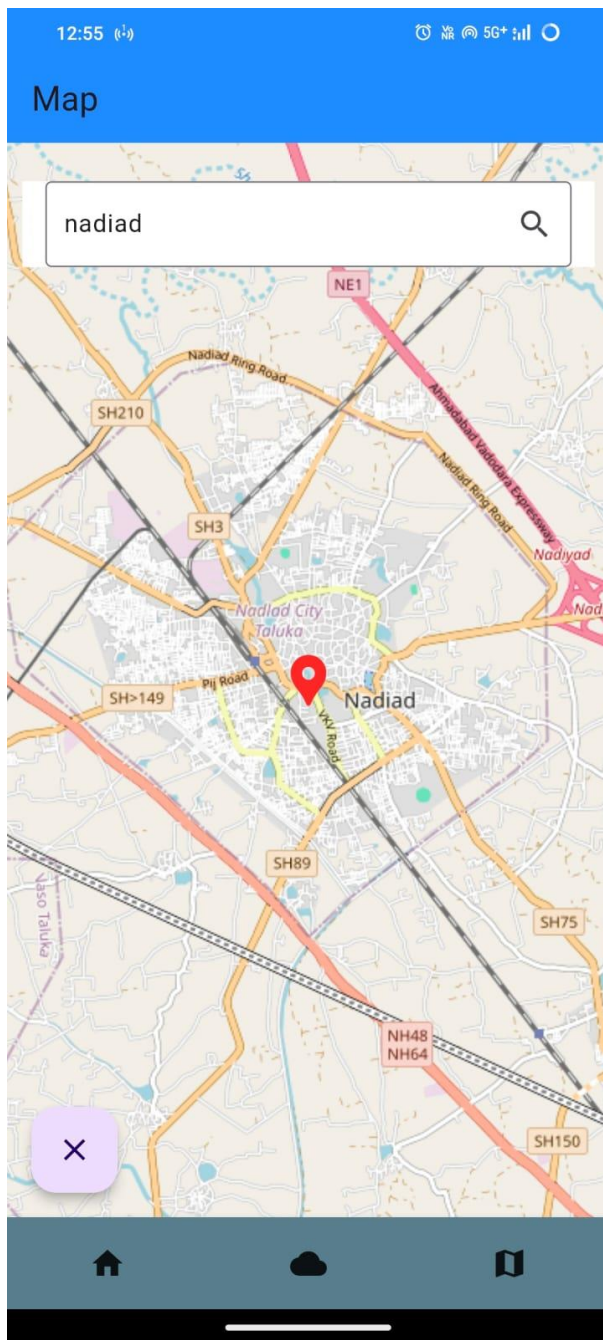
CHAPTER 6: Screenshots of your project output

Application Screenshots:









CHAPTER 7: Limitations of project

Limitations of project

1. Dependence on Internet for Certain Features

While the app provides offline access to critical information, some features, such as real-time weather updates and push notifications, require an active internet connection. In situations where network infrastructure is severely damaged, users may not receive timely updates or alerts.

2. Battery and Power Dependency

During prolonged disasters, users may face challenges in keeping their devices charged. Since the app relies on smartphones, its effectiveness diminishes when users are unable to charge their devices due to power outages.

3. Accuracy of AI Predictions

The AI-powered weather companion relies on available data and predictive algorithms to forecast disasters. However, the accuracy of these predictions is not foolproof, and sudden natural events may occur without sufficient warning, potentially leading to misinformation or delayed alerts.

4. Reliability of Third-party Services

The app relies on third-party APIs such as Twilio for SOS alerts and Google Cloud for weather data. Any downtime or disruptions in these services may affect the functionality of critical features like emergency notifications or real-time updates.

5. Limited Accessibility for Non-tech-savvy Users

Although designed to be user-friendly, the app may present a learning curve for older or non-tech-savvy individuals. Users who are unfamiliar with smartphones or app-based technologies may struggle to effectively use the app during emergencies.

6. Scalability and Localized Resources

The availability of local resources such as shelters and medical centers in the app may be limited to regions with adequate data or partnerships. In remote or underdeveloped areas, the app may not provide comprehensive or up-to-date information about available disaster relief resources.

CHAPTER 8: Outcome

Outcome

1. Enhanced Disaster Preparedness

The app empowers users with real-time weather alerts and disaster forecasts, enabling informed decision-making and proactive measures to reduce risks associated with natural calamities.

2. Improved Emergency Communication

The SOS feature allows users to quickly send distress signals with their location, facilitating faster response times and potentially saving lives during critical situations.

3. Reliable Offline Access to Information

Essential information, including evacuation routes and emergency contacts, remains accessible without internet connectivity, ensuring users have the necessary resources even during communication network failures.

4. Critical Importance of User-Friendly Design

A user-friendly interface is vital, especially in high-stress scenarios. Simplifying navigation helps users quickly access information, enhancing the app's effectiveness during emergencies.

5. Challenges of Real-time Data Integration

The project highlighted the complexities of integrating accurate real-time data from multiple sources, to ensure effective disaster management.

6. Emphasis on User-Centered Design

Engaging users throughout development reinforced the importance of understanding their needs and incorporating regular feedback, leading to a more practical tool for addressing real-world challenges during disasters.

CHAPTER 9: Future Enhancement

Future Enhancements

1. Machine Learning Integration for Predictive Analytics

Implementing machine learning algorithms can enhance the AI weather companion's predictive capabilities. By analyzing historical data and trends, the app could provide more accurate forecasts and early warnings for potential disasters.

2. Multilingual Support

Adding multilingual support would make the app accessible to a broader audience, ensuring that non-English speaking users can also benefit from its features. This enhancement would be particularly valuable in diverse communities affected by natural disasters.

3. Community Engagement Features

Introducing community engagement tools, such as forums or social media integrations, could facilitate information sharing and support among users during disasters. This would foster a sense of community and enable users to exchange real-time updates about local conditions.

4. Integration with Smart Home Devices

Future versions of the app could connect with smart home devices to provide automated safety measures during emergencies. For example, smart thermostats could adjust temperatures during heatwaves, or security systems could activate when disaster alerts are issued.

5. Expanded Resource Locator

Enhancing the resource locator feature to include a wider range of services, such as food banks, mental health support, and community centers, would help users find more comprehensive assistance during and after disasters.

6. Real-time Collaboration with Emergency Services

Establishing partnerships with local emergency services for real-time updates could improve the app's reliability. This collaboration would allow users to receive immediate alerts and directives from authorities during a disaster.

7. User Feedback and Improvement Loop

Creating a system for ongoing user feedback and suggestions would enable continuous improvement of the app. Regular updates based on user experiences can help tailor features to better meet their needs and enhance overall usability.

8. Offline Navigation Assistance

Future enhancements could include offline navigation capabilities, guiding users to safe locations or shelters using GPS even without internet access, ensuring they can find help when connectivity is lost.

CHAPTER 10: Conclusion

CONCLUSION

The development of the femme secure app has been an enriching and transformative experience. Throughout this project, we have explored the complexities of creating a comprehensive tool designed to enhance community preparedness and response to natural disasters. This journey has provided us with invaluable insights and significantly improved our technical skills and problem-solving abilities.

Working on this app allowed us to collaborate with a diverse team of talented individuals, addressing real-world challenges and utilizing advanced technologies to create a solution that meets the unique needs of users facing emergencies. From implementing real-time weather alerts and an intuitive SOS feature to ensuring offline access to critical information, each aspect of the project has deepened our understanding of user-centric design and development methodologies.

The mentorship and support from experienced professionals have been crucial in our growth. Their guidance, constructive feedback, and unwavering encouragement have shaped our development approach and instilled a sense of confidence in our abilities. We have learned the importance of continuous iteration based on user feedback, ensuring that our app not only meets but exceeds user expectations in times of crisis.

Reflecting on our journey with the disaster management app, we feel a profound sense of accomplishment and gratitude for the opportunities and experiences it has afforded us. This project has equipped us with tangible skills and knowledge while igniting a passion for creating impactful solutions in disaster response and preparedness.

As we look to the future, we are excited to continue our work in this critical field, leveraging the insights and lessons learned during the development of the app to contribute positively to future projects. We are confident that the skills, experiences, and relationships forged during this project will serve as a solid foundation for our career aspirations in technology and disaster management.

REFERENCES

<https://docs.flutter.dev/>

https://pub.dev/packages/flutter_local_notifications

<https://www.cometchat.com/docs/extensions/flutter-push-notifications>

<https://www.courier.com/guides/flutter-notifications>

https://pub.dev/packages/google_maps_flutter

<https://fluttergems.dev/geolocation-maps/>

<https://www.youtube.com/watch?v=VPvVD8t02U8>

<https://www.youtube.com/playlist?list=PLr7P7lMIUTuukkflWTYrPgTwueG-BoUHN>

<https://www.youtube.com/playlist?list=PL9n0l8rSshSmNoWh4KQ28nJn8npfMtzcs>

<https://console.twilio.com/>

<https://developers.google.com/maps/documentation/javascript/get-api-key>

